5th H AH 2008

5th International Conference on

Adaptive Hypermedia and Adaptive Web-Based Systems

Hannover, Germany 29 July - 1 August 2008

Adaptive Collaboration Support

Workshop Proceedings

Organizers

Alexandros Paramythis, *Johannes Kepler University Linz, Austria*
Stephan Weibelzahl, *National College of Ireland, Dublin*

Workshop on Adaptive Collaboration Support

Alexandros Paramythis[1], Stephan Weibelzahl[2]

Institute for Information Processing and Microprocessor Technology (FIM), Johannes Kepler University Linz, Altenbergerstr. 69, A-4040 Linz, Austria
alpar@fim.uni-linz.ac.at

National College of Ireland Dublin, Mayor Street IFSC, Dublin 1, Ireland
sweibelzahl@ncirl.ie

This workshop tackled an emerging topic in adaptive hypermedia: Adaptive Collaboration Support. Traditionally, Adaptive Hypermedia Systems were mainly geared towards adaptation of content and its delivery. Classic techniques such as adaptive link annotation, adaptive fragment hiding or sequencing manipulate the content or the presentation of the content to suit the user's individual needs. Most methods work at the content level, too. Only recently, have attempts been made to exploit adaptive techniques in order to support user activities. A prominent case where activity support becomes vital is collaboration: when two or more people work together toward a common goal, there is usually no given content to be presented, but the participants engage in an activity-oriented process. This process can be supported or optimized in a number of ways. For example, systems might observe interaction patterns and suggest the creation of communication sub-channels (e.g., a chat-room for three participants) or recommend suitable collaborators. Application areas for Adaptive Collaboration Support include but are not limited to collaborative learning, synchronous and asynchronous on-line discussion, video conferencing, group decision support, etc.

The workshop aimed to answer the following question: How can adaptation be of benefit in modelling and supporting that process? This can be further subdivided into three main workshop themes addressed:

- Models: Which aspects of users and groups (and their activities) need to be modelled and can be inferred or observed in the interaction between users or between user and system in order to support collaboration?
- Adaptation Methods & Techniques: Which existing methods and techniques can be reused or tailored to activity-oriented collaboration?
- Adaptation Languages: How can collaborative activities be formally described? How to author adaptations to and of collaborative activities?

We thank the members of the programme committee for their valuable contributions.
Paul de Bra, Eindhoven University of Technology, The Netherlands
Alexandra Cristea, Warwick University, UK
Jon Dron, Athabasca University, Canada
Tom Gross, Bauhaus University Weimar, Germany
Judith Masthoff, University of Aberdeen, UK
Julita Vassileva, University of Saskatchewan, Canda
Gerhard Weber, University of Education Freiburg, Germany

Workshop on Adaptive Collaboration Support

An Exploration of Adaptive Collaboration Support: Abstract for Invited Talk

Judith Masthoff

University of Aberdeen
Scotland, UK
j.masthoff@abdn.ac.uk

Computer Science Researchers have worked for decades on supporting collaboration, most notably in the fields of Computer Supported Collaborative Work (CSCW) and Computer Supported Collaborative Learning (CSCL). In this presentation, we will explore what *adaptive* collaboration support may entail and what has been done on this so far.

There are several distinct forms of adaptive collaboration support that may be provided:

- *Group Formation*. A (large) group of people may need to be divided into teams. Support may be provided for deciding who collaborates with whom and, if needed, on which subtask. Such a decision may be based on the capabilities and personalities of the individuals, as available in their user models. This may involve a notion of group utility and constraint satisfaction. This is probably were adaptive collaboration support started, with Brusilovsky (1999) describing as *the goal* of adaptive collaboration support "to use system's knowledge about different students to form a matching group for different kinds of collaboration". Adaptive group formation based on individual models has a.o. been studied by (Read et al., 2006; Alfonseca et al., 2006).
- *Peer Matching*. An individual user may need help or somebody to delegate a task to. Support may be provided in the form of recommending a suitable partner, as is e.g. done in I-Help (Vassileva et al, 2003). Users may also decide based on information about potential helpers, like their reputation, or a view of peers' user models (such as open learner models, Bull & Britland, 2007). Trust may play an important role in peer matching (Masthoff, 2007).
- *Interaction Support*. Not everybody in a group may contribute equally and appropriately. Support may be provided for convincing members to make contributions (dealing with lurkers) and listen to others (dealing with conversation hoggers). This may involve modelling group interaction (e.g. Read et al, 2006; McLaren et al., 2006) and incentive mechanisms (e.g. Cheng & Vassileva, 2006; Harper, 2007; Farzan et al, 2008). In particular in asynchronous collaboration, members may get overwhelmed by the number of contributions provided in their absence. Support may be provided for focussing members' attention to the contributions most relevant to them (e.g. Dron & Masthoff, 2004).
- *Decision Support*. Group members may vary in opinions, preferences and knowledge. This can cause problems when decisions are needed for the group as a whole (e.g. about what action to take next). Modelling common knowledge between

group members has been studied by (e.g., Introne & Alterman, 2006; Suebnukarn & Haddawy, 2006). Adapting to groups based on individual user models has a.o. been studied by (Masthoff, 2004; Masthoff & Gatt, 2006) and negotiation between group members by (Jameson, 2004).

In this presentation, we will also attempt to discuss what the main outstanding research questions are in this area, and how this area may benefit from work in affective computing and persuasive technology.

References

Alfonseca, E., Carro, R.M., Martín, E., Ortigosa, A. and P. Paredes: 2006, 'The Impact of Learning Styles on Student Grouping for Collaborative Learning: A Case Study', *UMUAI, 16*, 377-401.

Brusilovsky, P.: 1999, 'Adaptive and Intelligent Technologies for Web-based Education', *Künstliche Intelligenz 4*, 19-25.

Bull, S, and M. Britland: 2007 'Group Interaction Prompted by a Simple Assessed Open Learner Model that can be Optionally Released to Peers', Workshop on Personalisation in learning environments at individual and group level, UM07, Corfu.

Dron, J, and Masthoff, J: 2004, 'We are, of course, bored with course boards: An online alternative to formal course meetings'. Italics e-journal, 3, 2.

Farzan, R., DiMicco, J., Millen, D., Dugan, C., Geyer, W. and E. Brownholtz: 2008., 'Results from deploying a participation incentive mechanism within the enterprise'. CHI 2008, 563-572

Harper, M: 2007, 'Encouraging Contributions to Online Communities with Personalization and Incentives'. *User Modeling* 2007 Doctoral Consortium.

Harrer, A., McLaren, B.M., Walker, E., Bollen L. and J. Sewall: 2006, 'Creating Cognitive Tutors for Collaborative Learning: Steps toward Realization', *UMUAI, 16*, 175-209.

Introne, J. and R. Alterman: 2006, 'Using Shared Representations to Improve Coordination and Intent Inference', *UMUAI, 16*, 249-280.

Jameson, A.: 2004, 'More Than the Sum of Its Members: Challenges for Group Recommender Systems', *Proceedings of the International Working Conference on Advanced Visual Interfaces,* Gallipoli, Italy, 48–54.

Masthoff, J.: 2004. 'Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers'. *UMUAI, 14*, 37-85.

Masthoff, J. and A.Gatt: 2006. 'In pursuit of satisfaction and the prevention of embarrassment: Affective state in group recommender systems'. *UMUAI, 16*, 281-319.

Masthoff, J.: 2007. 'Computationally modelling trust: An exploration.' Proceedings of the SociUM workshop associated with the User Modeling conference, Corfu, Greece.

Read, T., Barros, B., Bárcena, E. and J. Pancorbo: 2006, 'Coalescing Individual and Collaborative Learning to Model User Linguistic Competences', *UMUAI, 16*, 349-376.

Suebnukarn, S. and P. Haddawy: 2006, 'Modeling Individual and Collaborative Problem-Solving in Medical Problem-Based Learning', *UMUAI, 16*, 211-248.

Vassileva, J., McCalla, G., and Greer, J.: 2003, 'Multi-Agent Multi-User Modeling in I-Help', UMUAI, 13, 179-210.

Systems for Adaptive Collaboration Scripting: Architecture and Design

Anastasios Karakostas, Stavros Demetriadis

Informatics Department, Aristotle University of Thessaloniki, Greece
POBOX 114, Postal Code 54124, Thessaloniki, Greece
{akarakos, sdemetri}@csd.auth.gr

Abstract. Although adaptation techniques have been used in the computer-supported collaborative learning (CSCL) area, there has not been such effort so far in relation to collaboration scripts. Collaboration scripts are didactic scenarios that guide and support groups of learners in collaborative learning. Adaptive collaboration scripting is the idea that collaboration scripts can be adapted during run time in several of their aspects to provide learning experiences optimized both for the individual learner and the group characteristics. Systems that will support this kind of adaptation should be based on adaptation models that consider both the user characteristics (at individual and/or group level) and the script features. In this paper we propose (a) a generalized system architecture for learning environments that adaptively support learners during scripted collaboration sessions and (b) a design case study of a web-based system for supporting the adaptive operation of a "pyramid" type collaboration script.

1 Introduction

Computer supported collaborative learning (CSCL) systems have already embodied characteristics from adaptive and intelligent Web-based educational systems (AIWBES). There have been two major approaches: (a) adaptive group formation and peer help and (b) adaptive collaboration support [1]. In the former approach belong systems that perform group formation based on users' personal features and preferences [2] or users' learning records (such as interaction style) obtained during an individual learning phase [3]. Systems, in the latter approach, implement group performance modeling based on group's both learning and social characteristics [4] in order to provide interactive support during the learning process. However, no explicit effort so far has been reported for embodying adaptation techniques in systems for scripted collaboration. Collaboration scripts are didactic scenarios that "facilitate social and cognitive processes of collaborative learning by shaping the way learners interact with each other" [5]. A script provides specific instructions for small groups of learners about the activities that need to be executed, when and by whom they need to be executed in order to enhance individual knowledge acquisition. The increased interest on scripted collaboration methods has motivated recently several efforts for the formalization of collaboration scripts and the development of computer-based environments for supporting scripted collaborative learning [5].

In this work we elaborate on the idea of adaptive scripting by presenting (a) a concise relevant conceptual framework, (b) a generalized system architecture for building adaptive collaboration scripting systems, and (c) a case study on the design a web-based system for supporting the adaptive operation of a “pyramid” type collaboration script.

2 Adaptive Collaboration Scripting

Adaptive collaboration scripting is, in general, the idea that computer–based environments for scripted collaboration can operate in an adaptive mode, in order to considerably improve the collaborative learning experience. In order to implement characteristics of an adaptive operational mode in such environments we should analyze the structural features of a script. These features impose on the learning activity both “intrinsic” and “extrinsic” constraints [6].

By “intrinsic” we refer to features that give script its specific pedagogical value and can not be changed (or adapted) in any way. Dillenbourg and Tchounikine [6] argue that these features set up the limits of script flexibility, that is, they define what can not be changed within the scripted learning experience if the pedagogical purpose of the script is to remain intact. By contrast, “extrinsic” constraints refer to script’s aspects that can be changed/adapted in order to provide room for learners/tutors to adapt the learning experience to their own preferences and characteristics. We suggest that extrinsic constraints can belong to either of two categories: (a) “Non-pedagogical”, that is constraints without any pedagogical relevance. These constraints can be altered by the teacher and/or the students simply to make the script to better accommodate the conditions of the specific implementation (for example, extending the duration of a phase because of a learner’s temporal inability to meet a deadline). (b) “Pedagogical” constraints that can (should) be adapted in order to provide a more profound and productive learning experience to the specific group of learners. An example of adaptation in a “pedagogical” constraint could be repeating a script phase in order to help novice learners better understand the material or the collaborative process. Another example could be adapting the way that a system performs group formation according to student models. From our perspective, therefore, the conceptual basis of adaptive collaboration scripting is a representation of script pedagogical extrinsic characteristics. Each of these script features constitutes a different “dimension” for adaptive behaviour. Consequently, a system for supporting adaptive collaboration scripting should be able to perform learner and/or group related adaptations in any of those dimensions – at least in principle.

3 A Proposed System Architecture

A proposed architecture for building systems that support adaptive collaboration scripting is shown in Figure 1. The architecture consists of three main layers: the *storage* layer which includes all the necessary elements to perform adaptation, the *group management* layer which constructs groups and manages group characteristics and, finally, the *runtime* layer which is the closest layer to the learner and contains all the interactions with him/her.

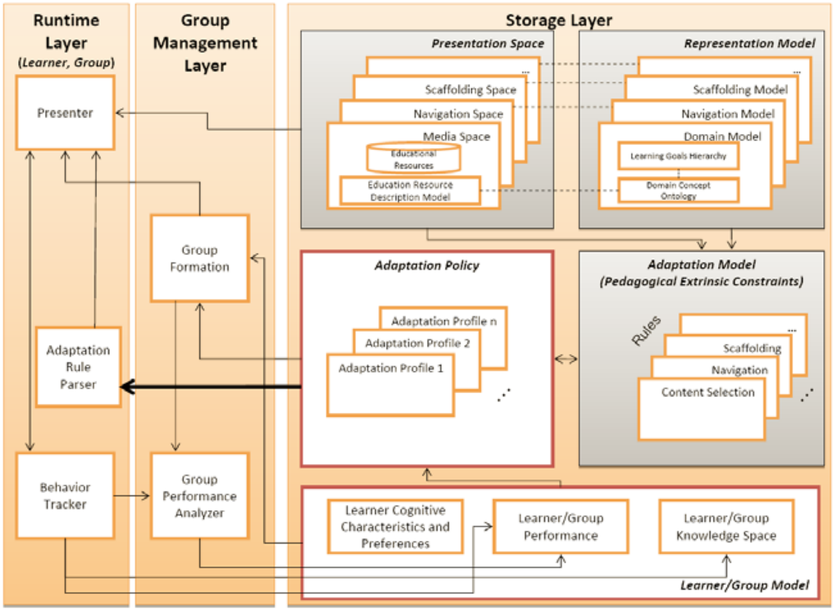

Fig. 1. Architecture of Adaptive Collaborating Scripting Systems (adapted from [7])

The storage layer primarily consists of the *learner/group model* component, which includes learner's cognitive characteristics and preferences about the learning process or the domain. It also records the learner/group performance during the educational procedure and the actual learner/group knowledge space. Learner/group model is input to the *adaptation policy* component which actually provides the representation of the script pedagogical extrinsic features and enables system users to activate adaptations on any of these aspects, by editing a respective *adaptation profile*. Adaptation profiles (there should be at least one) can be conceptualized as a set of choices predefined by the tutor, on the adaptation rules that the system should activate and the aspects of the user/group model that should be considered. Adaptation policy communicates with the *adaptation model* component which stores the rules for describing the kind of adaptation that is activated during system runtime.

Presentation Space contains the respective presentation modules, that is, all necessary information to be presented onscreen when a specific adaptation rule is enacted. For example media space contains all forms of learning material that can be presented by the system when performing content adaptation. Each module in the presentation space (media, navigation, scaffolding, etc.) has its counterpart in the *representation model,* which provides a representation of its structure, relevant to a learning goal hierarchy. If, for example, the system identifies three levels of learner/group competence (novice, advance, expert) the relationship between these levels is represented in the scaffolding model (within the representation model component). The *group management layer* consists of the *group formation* and the *group performance analyzer* modules. Group formation module defines groups of students, based on available information from the user/group model and the active adaptation profile. Adaptation profile is responsible for providing rules for group formation (for example forming homogeneous groups of three based on their prior knowledge). Group performance analyzer analyzes the performance of the group and sends the filtered data to group performance module of user/group model. It has also as input the group structure from the group formation module in order to identify the type of group it analyzes. Finally, the *runtime layer* consists of a behaviour tracker, an adaptation rule parser and the presenter.

4 A Design Case Study: the Pyramid Script

The design case study concerns a pyramid type collaboration script to support students when engaged in case-based learning (CBL). Generally in a learning activity of the pyramid-type collaboration each participant works first individually studying a problem (or any learning material) and then participates in a workgroup of a gradually increasing size, to collaboratively process the material from a certain perspective. The "pyramid" script for computer based learning consists of four phases: (a) an individual study of one or more cases, (b) a collaboration phase in a small group, (c) a collaboration phase in a larger group communicating through asynchronous discussion tools, and (d) a debriefing phase conducted in the classroom. To design an adaptive computer-based system for implementing the pyramid script, we deem as appropriate to consider at least two adaptation approaches: (a) group formation methods based on the individual student models, and (b) adaptation techniques based on group performance model.

In typical "pyramid" script implementations small groups (in the second phase) are formed randomly. We suggest that group formation techniques can be used at this point to enable adaptive group formation based on pedagogically sound hypotheses. The group formation process can be based on (a) the students' prior knowledge on didactic model (in our case study the case-based learning), (b) the students' prior knowledge on the domain and (c) the students' background. Furthermore the group formation process relies on the tutor's "policy" on how to form the groups in order to facilitate learners' interactions. For example a group formation phase, based on the students' prior knowledge on didactic model, can put in the same group learners with different level of prior knowledge to facilitate knowledge dissemination.

In the second script-phase, group performance modelling techniques can be used to keep track of various group performance parameters. The group performance model could have both learning and social aspects, such as measuring group members' participation and motivation. Based on this group model the system can take decisions (criteria control) on the adaptation rules that should be enacted. If a criteria control is not satisfied (i.e., group members participation is at low level) the system might automatically activate an adaptation rule (for example, activating a scaffolding/motivating mechanism) (automatic adaptation) or inform the teacher to take any appropriate measures (semi-automatic adaptation). The user/group model and the adaptation rules are emerged from the pyramid script pedagogical extrinsic constraints (an extended list of extrinsic and intrinsic constraints can be found in [8]).

5 Conclusions and Future Work

In this paper we have argued in favor of an adaptive approach to scripted collaborative learning as a method for enhancing the learning interactions among the students during the learning experience. We have suggested that for building systems for adaptive scripting one need first to distinguish between intrinsic and extrinsic script constraints and develop a respective computerized script representation. We have also presented a generalized system architecture for learning environments that will adaptively support learners during scripted collaboration sessions and a design case study of a web-based system for supporting the adaptive operation of a “pyramid” type collaboration script.

We are currently working on the development of a system for supporting students in adaptive pyramid-type scripted collaboration, identifying the script intrinsic/extrinsic constraints and construction adaptation rules that will be integrated to the system. In evaluating the system we will focus on assessing the students’ satisfaction and quality of learning compared to the non-adaptive treatment and also the cost/effort ratio of using the system from the instructor’s viewpoint

References

1. Brusilovsky, P., Peylo, C.: Adaptive and Intelligent Web-based Educational Systems. International Journal of Artificial Intelligence in Education, 13, pp.156–169 (2003)
2. Carro, R.M., Ortigosa, A., Schlichter, J.: Adaptive Collaborative Web-based Courses. In: Cueva, J.M., González, M., Joyanes, L., Labra, E. and Paule, M.P. (eds.) Web Engineering. Lecture Notes in Computer Science. pp. 130-133. Springer (2003)
3. Quignard, M., Baker, M.: Favouring modellable computer-mediated argumentative dialogue in collaborative problem-solving situations. In: Proceedings of the 9th International Conference on AI in Education, pp. 129– 136. IOS Press, Amsterdam (1999)
4. Vizcaino, A., Contreras, J., Favela, J., Prieto, M.: An adaptive, collaborative environment to develop good habits in programming. In: Proceedings of the 5th International Conference on Intelligent Tutoring Systems, pp. 262-271. Montreal, Canada (2000)
5. Kobbe, L., Weinberger, A., Dillenbourg, P., Harrer, A., Hämäläinen, R., Fischer, F.: Specifying computer-supported collaboration scripts. International Journal of Computer-Supported Collaborative Learning, 2, 2-3, pp. 211-224. (2007)
6. Dillenbourg, P., Tchounikine, P.: Flexibility in macro-scripts for computer-supported collaborative learning. Journal of Computer Assisted Learning, 23, 1, pp. 1-13 (2007)
7. Karampiperis, P., Sampson, D.: Adaptive Learning Resources Sequencing in Educational Hypermedia Systems. Educational Technology & Society, 8, 4, pp. 128-147 (2005)
8. Demetriadis, S., Karakostas, A.: Adaptive Collaboration Scripting: A Conceptual Framework and a Design Case Study. In: Proceedings of the Second International Conference on Complex, Intelligent and Software Intensive Systems, pp. 487-492. Barcelona, Spain (2008)

A Report on Participatory Workshops for the Design of Adaptive Collaborative Learning

Kevin Muise, Karen Tanenbaum, Ron Wakkary and Marek Hatala

School of Interactive Arts and Technology
Simon Fraser University
250 - 13450 – 102nd Avenue
Surrey, BC, Canada, V3T 0A3
{kmuise, ktanenba, rwakkary, mhatala} @ sfu.ca

Abstract. This “work-in-progress” paper discusses the process of conducting participatory workshops during the design of an adaptive, collaborative learning system. We outline our methods for exploring group interaction, collaboration, and adaptive learning in an iterative series of workshops. We conclude with a discussion of how the results of these workshops have influenced our ongoing work in designing an adaptive guide system for family groups in the museum.

Keywords: adaptive systems, collaboration, learning, participatory design

1 Introduction

Our current research project aims to create a museum guide supported by an adaptive group user model that supports different learning and interaction styles for a tangible-user interface system. Family groups will collaborate using various multi-modal tangible devices to collect information about artifacts in a museum and later combine the collected artifacts in order to construct a mutual understanding of their experience in the museum. As part of the design process for this system, we studied the patterns of collaboration in pairs of people engaged in a series of playful learning tasks built around “treasure-hunt” and “puzzle” themes. The exploratory workshop results guided the design of a group interaction framework and an adaptive model for task assignment. This work-in-progress paper discusses the process of conducting the iterative workshops and the impact of the findings on our design thinking.

Participatory workshops are frequently used to explore design ideas in ways that are simultaneously analytic and generative. Further, workshops allow for a focus on particular design situations within a larger design problem. When exploring complex interactions, workshops allow designers to explore specific facets of the interaction rather than tackling the whole scenario at once [5]. Workshops are developed iteratively, each one constructed as a response to the previous results and a refinement of the original investigation. Ehn suggests that game playing within participatory workshops helps create opportunities for designers to learn from participants, and thus the game becomes a tool for research [5]. In particular, the use of game pieces in participatory workshops has been widely discussed [1], and our workshops used a jigsaw puzzle to structure the activity. Previous research that employed jigsaw puzzles used them to investigate collaboration itself [3], whereas we use puzzles to better understand the effects of an adaptive task assignment model and to observe emergent collaborative patterns.

2 Participatory Workshops

In this report of our project, we describe three sets of workshops, each of which asked pairs of participants to assume fictional roles and accomplish tasks related to the roles. The workshops were conducted in rooms that had been "augmented" with paper tags that provided additional information on physical objects in the space. These paper augmentations were meant to mimic the information that would be available in the museum setting via the guide system. The participant tasks typically required moving around the space, collecting specific tags and returning them to a central location. Task assignments, or a selection of tasks to choose from, were given to participants by the workshop facilitator. Throughout the course of all three workshops, we varied the amount of collaboration that was required or encouraged by the activity and task structure, in order to see what patterns of interaction would emerge from the different design decisions. We also explored different ways of assigning tasks to participants to investigate the system could adapt to individual and group characteristics. The learning model used to structure the experience was Bloom's taxonomy: Remember, Understand, Analyze, Apply, Create and Evaluate [4], with each individual task designed to be at one of these levels.

Workshop 1: This workshop focused on determining if the participants would notice if tasks got more or less challenging as different methods of assigning tasks were used, and thus was primarily an exploration of *adaptivity*. We color-coded information tags and placed them on specific objects in each of the two rooms used for the workshop, instructing participants to only collect the color that they were assigned. The goal for the participants was to develop an understanding of the items by reading and collecting the information cards necessary to complete their assigned task, and then returning these cards to the facilitator. The tasks were designed to increase in difficulty along the first three cognitive levels of Bloom's taxonomy. If the participant returned the correct items according to the task, they were given a task on a higher level. If they returned items outside the range of appropriate answers, they were given another task at the same level.

The reaction to the task levels and assignment scheme was mixed. Although participants said they did learn about the objects in the rooms, they reported that all of the tasks seemed easy and they did not discern much difference between the different levels. We determined that ambiguous task descriptions played a large role in the answers the participants brought back and how they interpreted the tasks. In our observations of this workshop we saw that there was little group interaction, which we hypothesized was due to each participant being told to access information only from their assigned color. This limited the participant's ability to communicate and share information or help each other complete tasks. This workshop played an exploratory role in understanding collaboration and adaptivity.

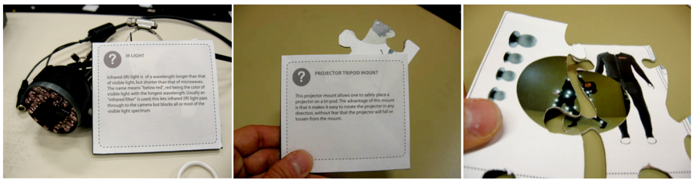

Fig. 1. Text tags and puzzle pieces used for workshops 2 and 3

Workshop 2: In our second workshop, we focused on *collaboration* and introduced a physical representation of a shared activity, which took the form of a puzzle (Fig. 1). Each puzzle piece had an image representing an item in the room, and the pieces were placed next to those items along with some explanatory text, as the tags had been in the first workshop. We abandoned the color-coded system that had inhibited group collaboration, allowing any participant to choose any puzzle piece. The participants therefore had to negotiate who needed the specific piece more, as there was only one puzzle piece for each item. The two participants had to work together to learn how to create an interactive installation that involved six separate components (cameras, projector, etc.). Each of the six component systems was a section of the puzzle containing between two to four pieces that when assembled correctly, created a shared composite image. Additionally, tasks were no longer assigned by a facilitator, but rather, each participant could select their own task. Each task corresponded to finding and assembling one of the 6 components. The puzzle itself could be constructed in multiple ways, with multiple pieces for one position in the puzzle. However, depending upon which puzzle pieces were combined to create a component, other sections could not be connected in a proper manner. Thus the participants had to negotiate how each individual section would be constructed to complete the puzzle.

In completing each task, the participants had two main approaches, i) looking for specific objects from a logical perspective, ii) going through the items in a trial-and error manner, checking for physical matches between the puzzle pieces. When an individual's task was completed, they would come back to the table and select another task, which would often place pressure on the other member who was still working on their task. This created a sense of competition between the team members, despite the activity being one of collaboration. In the post-task interviews, the participants noted that the puzzle helped them to reflect upon their tasks. We purposefully did not include an adaptive learning element in this workshop, as we wanted to focus how to structure a basic collaborative activity.

Workshop 3: In the third workshop, we used the same puzzle and activity structure, but changed the way in which the participants could combine them and select tasks, thus bringing together the *adaptivity* and *collaboration* based explorations of the previous workshops. The task descriptions were carefully adjusted to create a learning scale, where some activities were simple and others were more complex, based on Bloom's taxonomy. Learning from the first workshop, we paid closer attention to wording and potential ambiguities. The workshop provided fewer task choices for the participants than the second one, attempting instead to provide a level of challenge based on adapting to the participants' learning level. Also, instead of allowing a participant to choose another task once they were finished, we instructed them to help their partner to finish their task. In post-task interviews, each participant's reported sense of challenge corresponding to our intentions of manipulating the learning level by restricting their choices to tasks at a certain level.

3 Discussion and Conclusions

In our first workshop, little collaboration occurred when the participants were simply carrying out individual tasks, but it did provide us with insight in how to restructure the activity. Introducing the puzzle assembly helped make explicit the collaborative aspects of the workshop, despite it only being a representation for the more abstract learning goal. Achieving a good response to the adaptive task assignment also took several iterations as we learned how to gauge task difficulty as well as assess participant competency. Through conducting these workshops, we identified three distinct types of collaboration: 1) an individual working on a solo task that contributes to a shared goal, 2) an individual working on a solo task with a second person helping them out, and 3) two members working on a shared task for a shared goal.

In our third workshop, we observed all three of these collaborative behaviors at different times, and also achieved an adaptivity strategy that increased the challenge level appropriately for each individual while still maintaining group cohesion. Focusing the collaborative aspects by having participants assembling the puzzles at the table at the same time increased interaction and cohesion as well. As a final methodological observation, we have found that it is important to start simple in developing workshop structures for group collaboration, as it can be difficult to tease apart the different elements that are affecting participant behaviour and response.

Through developing these workshops, we were able to gain particular insights into group collaboration and how to structure our adaptive model to create an engaging learning experience for our users. This process showed us the need for a shared understanding of a group activity and goal as well as careful attention to individual competencies. We hope that this preliminary report can serve as an illustrative case study in the design of a complex adaptive, collaborative system.

References

1. Brandt, E. and Messeter, J.: Facilitating Collaboration through Design Games, Proc. Participatory Design Conference 2004, pp. 121-131, ACM Press, Toronto (2004).
2. Ehn, P.: Work-orientated Design of Computer Artefacts. Stockholm, Almquist and Wiksell International. (1988).
3. Johnson, H. and Hyde, J.: Towards Modeling Individual and Collaborative Construction of Jigsaws Using Task Knowledge Structure (TKS), ACM Transactions on Computer-Human Interaction, vol 10, iss. 4, pp. 339-387, (2003).
4. Starr, C. Manaris, B. and Stalvey, R.: Bloom's Taxonomy Revisited: Specifying Assessable Learning Objectives in Computer Science. In: 36th SIGCSE technical symposium on Computer science education, pp. 261-265, ACM press, Portland (2008).
5. Wakkary, R.: Framing Complexity, Design and Experience: A Reflective Analysis, Digital Creativity, vol. 16, iss. 2, pp. 65-78, (2005).

Adaptive Support for Collaborative Learning with IMS Learning Design: Are We There Yet?

Alexandros Paramythis

Institute for Information Processing and Microprocessor Technology (FIM)
Johannes Kepler University Linz
Altenbergerstraße 69, A-4040 Linz, Austria
alpar@fim.uni-linz.ac.at

Abstract. This paper examines ways in which on-line collaboration in general, and on-line collaborative learning in particular, can be supported using existing and novel adaptivity techniques. In that context, a set of requirements that need to be met for providing adaptive support for collaborative e-learning is formulated. The IMS Learning Design specification is then assessed against these requirements, to determine whether it can serve as a sound basis for implementing the aforementioned types of adaptive support. The paper concludes that, although this specification is a promising one in this respect, it still lacks several features that would be vital in adaptively supporting collaborative learning.

Keywords: adaptivity, collaboration, e-learning, CSCL, IMS LD

1 Introduction

The proliferation of Internet- and Web- technologies in the last two decades has brought about major changes in practically all facets of human activity. This trend has already had tremendous impact in the field of education, where new teaching and learning paradigms have been established under the general “umbrella” of e-learning (or, to use an equally popular term, open and distance learning). New technologies have allowed for much richer learning experiences, as well as for the elimination of temporal and geographic barriers traditionally associated with learning activities. The strides achieved, however, have been accompanied by an inevitable decrease in the amount of face-to-face contact between instructors and learners, and especially between learners themselves, which has created new hurdles in the education process.

It is widely acknowledged that a large part of success of the learning process lies with the opportunities of learners to interact with others: groupwork, exchanging ideas, and helping each other (thereby learning themselves) are standard “classroom” practices. With limited real-world contact, learners have limited means for discovering other learners’ capacities, skills, interests, strong and weak points, disposition towards teamwork, willingness to help, learning progress etc. Without such knowledge, learners cannot make informed decisions about everyday learning tasks like: whom to direct a question to; which person(s) have the complementary skills required to put together a group that can effectively work on a given task; when to contact them; etc.

Fostering exchanges between online students can also lead to social cohesion, and, more specifically, to a psychological sense of community [1]. The later has been shown to be a major factor in attaining study-related satisfaction [2], achieving successful learning outcomes [3], preventing student burnout [4], and decreasing dropout rates [5]. Research about distance-learning has also revealed that interactions among students and instructors increase the effectiveness of learning [6] and is beneficial both to individuals and to institutions [7].

This paper examines ways in which on-line collaboration in general, and on-line collaborative learning in particular, can be supported using existing and novel adaptivity techniques. To this end, a set of requirements that need to be met for providing adaptive support for collaborative e-learning will first be discussed. We will then examine which of these requirements are met by the IMS Global Learning Consortium's Learning Design specification (IMS LD [8]). IMS LD is specifically targeted because it is, at the moment, the only widely known and used specification that provides a language for modeling group learning activities. The conclusion of the presented analysis is that IMS LD needs to evolve further before it can be used for adaptive collaboration support and some of the required evolution steps represent a considerable departure from the current form of the specification.

2 Adaptive Support for Collaborative Learning

In the context of this paper, adaptive collaboration support refers to adaptive support in learning processes that involve communication between multiple persons (and, therefore, social interaction), and, potentially, collaboration towards common objectives [9]. Such support is intended to contribute towards intelligent and automated approaches to online learning that are in line with modern learning theory, which increasingly emphasizes the importance of collaboration, cooperative learning, communities of learners, social negotiation, and apprenticeship in learning [10].

The theme of adaptive collaboration support lies at the crossroads of two areas of work that have been evolving independently until now: Computer-Supported Collaborative Learning and Adaptive / Intelligent Learning Systems.

Computer-based and computer-supported cooperative / collaborative systems have emerged to enable people to perform tasks and carry out activities synergistically, over temporal and geographic distances [11]. Computer-Supported Collaborative Learning (CSCL) arose from research on CSCW and emerged as a separate field of study in the early 90s. Put briefly, CSCL is focused on how collaborative learning supported by technology can enhance peer interaction and work in groups, and how collaboration and technology facilitate sharing and distributing of knowledge and expertise among community members.

The field of adaptive e-learning has grown out of work on Intelligent Tutoring Systems, amalgamated with progress in the area of user-adaptive systems, as applied in computer-supported e-learning. Current systems that adaptively support collaboration maintain models for a large number of users, and use the information in these models to facilitate the establishment of collaboration activities, as well as to support ongoing collaboration itself. User modeling has been applied in connection with several (partially overlapping) types of collaboration: in computer-supported learning environments (see, e.g., [12]); as a means to provide adaptive social awareness support (see, e.g., [13]); as way of providing "intelligent help" for complex tasks (see, e.g., [14]); putting human expert into the loop as way of avoiding some of the difficulties associated with fully automatic adaptive help systems; in environments for computer-supported cooperative work within organizations (see, e.g., [15]); etc. Adaptive learning environments are increasingly being made available on the web, with representative examples including ELM-ART [16] and SQL TUTOR [17].

Adaptive techniques that have been used in the area of collaboration support can be broadly categorized into ones that address the establishment of collaboration in the first place (e.g., group establishment) and ones that support the collaboration process itself [18]. Although these categories are by no means mutually exclusive, they do exhibit significant differences in their adaptation determinants and constituents. We will outline these in the rest of this section, and use them later on as a guide for assessing the sufficiency of IMS LD for implementing adaptive collaboration support.

The analysis that follows focuses specifically on *collaboration* support. It is assumed, however, that information regarding *individual learner activities* is also available to the adaptation components and algorithms where this might be needed (e.g., it is assumed that systems maintain a learner model per individual learner).

Adaptive Support for the Establishment of Collaboration

Systems in this category are typically based on the learners' personal- and learning-characteristics and preferences, either explicitly stated by the users themselves, or observed / inferred during their interaction with the system [19] [20]. A lot of these adaptation determinants can be readily represented in user and learner models following existing practices and utilizing existing specifications such as the IMS Learner Information Package specification [21]. This is in large part due to the fact that this category of systems does not need to explicitly model the activities and performance of groups, but rather to facilitate their formation; thus, modeling only individuals is an essentially viable approach. At the same time it can be assumed that at least part of the information stored in the user- / learner- profiles is automatically derived by the system through observation. Examples of cases where this would be desirable include situations where: manual maintenance of the information by the users themselves would be tedious (e.g., recording their progress through learning materials); objective assessments must be derived (e.g., performance of learners in a test); etc.

More recent approaches to adaptively supporting the establishment of collaboration go further than described above, to take into account: (a) historical activities of learners in collaboration contexts; and (b) the learners' current engagement in collaborative activities, from long-term asynchronous ones, to short-term synchronous ones. This allows for the implementation of a wide range of adaptation strategies, from ones that "couple" users based on their propensity to collaborate and participativity, to ones that take into account a learner's current collaboration load and availability (including instantaneous load, such as when the user is participating in a live audio session).

The type of information described above cannot readily be stored in traditional user- or learner- profiles, unless it has already been processed and summarized so that it can be expressed in the form of user properties. Additionally, because of the sheer amount of involved information, it is not reasonable to expect users to explicitly provide the related information to the system, especially since the system is evidently "present" when the related activities take place (but the users should maintain control over the system's inferences in most if not all cases).

Based on the above, we can identify a number of high-level requirements as far as adaptation in this category is concerned:

A1. Capability to automatically collect / infer and model *user- and learner- profile data* of individual learners, and provide access to said data for adaptation algorithms.

A2. Capability to automatically collect / infer and model *collaboration activity data* for individual learners, and provide access to said data for adaptation algorithms; this may also entail observing the *activity of groups as collective entities* in the process. Activity data can be modeled in many alternative ways. This requirement is not prescriptive as to the modeling approach used, as long as the system can provide at least aggregate information about the activities of individual learners, grouped by activity category and learning context (e.g., messages a student posted to a project-related forum, as a percentage of total messages posted in that forum for the duration of the project).

A3. Capability to represent and employ *algorithms / strategies* that govern how learner information is used to *identify appropriate collaboration partners*. Note that the act of identifying collaboration partners is here considered separate from how the resulting information is used further by the system. In existing systems, it is typically used directly for partner recommendation. Many alternative uses are possible though: determining the "fit" of teams established directly by their members; identifying missing skills in a group; making a prognosis about potential problems a group may encounter based on its members' characteristics; etc.

A4. Although not an absolute requirement, it would also be desirable for the system to allow for *alternative policies for / approaches to group initiation*. Examples of such policies include: assign a peer to assist a learner that is assumed to be encountering difficulties while taking an online knowledge practice test; cluster course participants into groups to tackle a question posited in a class setting; etc.

Adaptive Support during the Collaboration Process

Systems that support the collaborative learning process itself need first and foremost to model the said process and the performance of groups within it, based on both learning and social characteristics [22]. Although work to introduce adaptivity in this area is limited at the moment (arguably due to the non-trivial effort involved), it is rather straightforward to identify the constituents and determinants of adaptation, if we accept that, in the ideal case, the adaptive system should be able to take over the role that a human facilitator might play in the process.

Starting then with the adaptation constituents, at the first level, the system would need to be able to observe the collaboration activities of learners. However, in this case, such observations need to be a lot more elaborate than what was described above. Specifically, the system would need to be able to identify / discern activities with higher accuracy; it would also need to identify the services and artifacts that these activities involve and their state (e.g., a document being jointly edited by group members on a wiki). At a second level, the system should be able to: (a) determine whether any expressed constraints about the process are observed by the group as a whole, or by the individuals within it, and (b) identify patterns of group activity in a semantically meaningful way so that they can be acted upon if and when necessary.

Further to the above, and in contrast to systems only concerned with facilitating group formation, in this case we also need to address the modeling of groups themselves as multi-faceted entities. Properties such as the group's creation policy, creation time, termination policy, assets available to its members, roles of the members, etc., all need to be explicitly represented and can be used as the basis for adaptation. Additionally, the system may need to maintain more elaborate group models, as is often for example the case in group recommenders [23], where collective properties of the group's members also need to be maintained.

Moving on to adaptation constituents, and carrying on with the analogy of an equivalent to an external to the group human facilitator, we can identify several aspects of the collaboration process where a system might be able to intervene: varying the group size; recommending or assigning (changes in) roles for participants; modifying the activity structure (e.g., by adding / removing / reordering tasks); determining the availability of elements (including activities, services and artifacts); etc. The challenge here, of course, is not to pick out the aspects of collaborative activities that can be modified, but rather: (a) to make sure that such changes can be made fully at run-time, and (b) that the changes made do not have detrimental effects on the learning process, or the activities that have already been completed or are under way.

Based on the above, we can derive another list of high-level requirements for adaptation in this category of collaborative learning support:

B1. Capability to maintain *models of groups*, including collective properties of the groups' members, automatically collect / infer information for these models, and provide access to them for adaptation algorithms.

B2. Capability to maintain *models of group activities*, including the roles of participants, the services used, the artifacts produced, etc.

B3. Capability to *guide the collaboration process*, using the aforementioned models of group activities.

B4. Capability to automatically collect / infer and model *collaboration activity data* of group members, including services used and artifacts used and / or produced, and provide access to said data for adaptation algorithms. This requirement is an extension of requirement A2, in that it dictates not only aggregate information about the activities of a group and its members to be available, but rather more detailed access to individual activities and groupings thereof.

B5. Capability to *identify group activity patterns in a semantically meaningful way* (e.g., patterns that may indicate conflicts amongst team members)

B6. Capability to represent and employ *algorithms / strategies* that govern how collaboration information is used to *identify appropriate interventions*

B7. Support for enabling the above *adaptation algorithms to modify any aspect of the collaboration process* (including aspects of its participants, and other participating entities). Such support may be constrained, and will typically also involve assessing the *validity* of the resulting process specification if the requested modifications were to be applied.

3 The IMS Learning Design Specification

The IMS Learning Design specification [8] evolved out of the Educational Modelling language (EML) developed by the Dutch Open University OUNL [24]. IMS LD is a learning process modeling language, fashioned on the theatrical play metaphor (e.g., with plays, actors, roles, etc.), and intended to formally describe any design of teaching-learning processes for a wide range of pedagogical approaches [24][25].

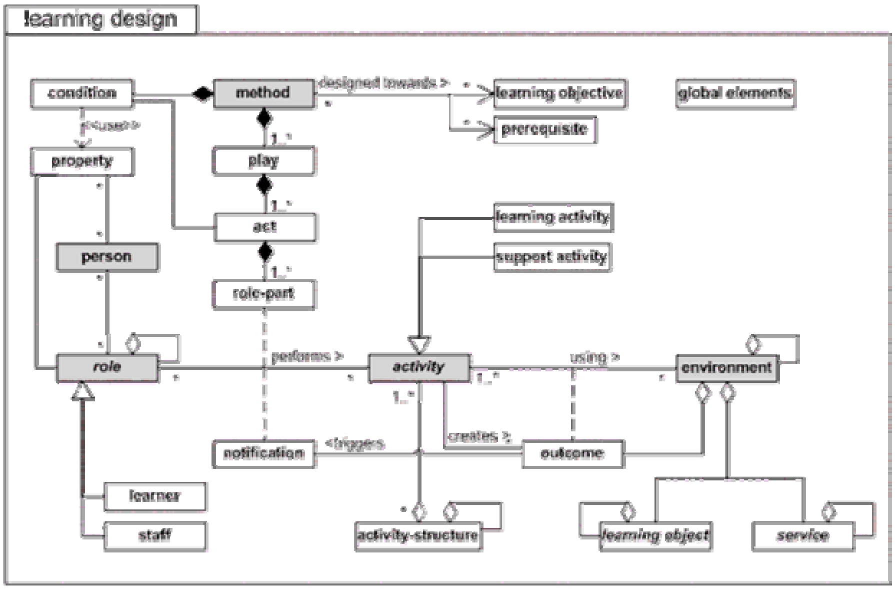

Fig. 1. Conceptual model of overall Learning Design (from [8])

The specification consists of three levels [8][26]. Each level itself provides specific features to the educational information embodiment, called the Unit of Learning. Level A provides method, plays, acts, roles, role-parts, learning activities, support activities and environments; Level B provides properties, conditions, calculations, monitoring services and global elements; and Level C provides notifications. Every level is built on the previous one. Level A is the main part of the specification, and forms the basis of any Unit of Learning. Level B adds powerful features to create more complex e-learning lesson plans. And, Level C provides an activity oriented triggering system. Fig. 1 provides an overview of the IMS LD conceptual model that results from the combination of all three layers.

Although the basic structure is provided by Level A, it is actually the elements of Levels B and C which provide the necessary mechanisms and affordances for adaptation, as they combine properties with conditions and other features that encourage and make more flexible the content and the learning flow [26].

According to the literature, IMS LD can support six main types of adaptation [27]: Learning flow based, content based, interactive problem solving support, adaptive user grouping, adaptive evaluation and changes in run-time (although, as we will see in the next section, the capacity of the specification in terms of adaptive user grouping can be strongly contested).

It should be noted that this specification is not intended to be used in isolation. IMS LD can be closely integrated with, or often relies on, other IMS-issued specifications. For instance, IMS LD does not directly support the representation of learning / skill tests – but can integrate them through the IMS Question and Test Interoperability specification [28]; the case is similar for maintaining learner profile information through the IMS LIP specification [21].

The IMS LD specification is targeted in this analysis for a number of interrelated reasons. Firstly, widely used Learning (Content) Management Systems (LMS) are very likely to support at least a selection of e-learning standards; at the same time, they are rather unlikely to include any custom provisions or mechanisms exclusively intended to support adaptivity [29]. Therefore, if one is aiming at wide-spread support for adaptivity in the context of collaborative e-learning, one should channel related research efforts through appropriate standards. Given that IMS LD is the only e-learning specification at the moment that addresses the topic of collaborative learning, and has gained traction with major LMS, it was felt that it constitutes the most effective vehicle for ensuring future complying systems have the features in place that would allow for adaptive collaborative e-learning to become a reality.

4 Adaptive Collaboration Support Using IMS LD

IMS LD has been acknowledged as a language with strengths in specifying *personalized learning* and *asynchronous cooperative learning*. However, IMS LD provides insufficient support to model group-based, synchronous collaborative learning activities [30]. Caeiro et al. [31] criticized IMS LD regarding CSCL purposes and suggested a modification and extension of the specification. The suggested changes are, however, restricted to the role- and method- parts of the specification. Hernandez et al. [32] suggested adding a special type of service, called “groupservice” to extend the capacity of IMS LD. It has been argued that such an extension at service level, rather than at activity level, cannot appropriately capture the characteristics of collaborative learning activities [30].

In the context of research examining IMS LD as a language in which CSCL scripts[1] can be formally expressed, Miao et al. [30] identify weaknesses of the specification in the following areas: modeling groups, modeling artifacts, modeling dynamic features, modeling complicated control flow, and modeling varied forms of social interaction. We will summarize and examine these in turn, along with other IMS LD problems and omissions that are more specific to adaptivity in the context of collaborative e-learning, to determine the extent to which they affect adaptive collaboration support prerequisites.

– *No built-in support for modeling groups*: IMS LD does not provide primitives for directly representing groups of learners. In some simple cases, groups may be pos-

[1] According to O’Donnell & Dansereau [33] a collaboration script is a set of instructions specifying how the group members should interact and collaborate to solve a problem.

sible to represent using roles and custom global properties[2]. This is not a viable solution in all cases though, as it requires an exhaustive enumeration of roles and sub-roles (to represent groups and sub-groups) at design time. Another problem the system cannot explicitly identify groups / sub-groups at run-time, which severely limits the system's capacity to support inter- / intra- group collaboration appropriately. A related problem is that, in IMS LD, roles are assigned to persons before running a unit of learning and remain unchanged within the life cycle of the run. As far as adaptivity is concerned, because it means we can't really add a person to a group dynamically, and it also means we can't dynamically change the role of a person, which removes a very important constituent from the adaptation "arsenal". In fact, as observed in [27], once a run starts, we can only add and remove users from roles, .but this is about all the flexibility available.

– *No built-in support for modeling artifacts*: A second major deficiency of IMS LD is in modeling artifacts. In learning processes, actors use and generate artifacts such as a vote, an answer, an argument, or a design. In IMS LD, an artifact can only be modeled as a property of the person / role / etc., which creates the artifact. In fact, attributes of the artifact (like type, status, creator, contributors, etc.) can also only be expressed as properties. Additionally, to ensure the validity of the representation, one would have to write elaborate rules governing the said properties and their values with very poor reusability. It's also worth mentioning that IMS LD doesn't support any array-like data structure, which also complicates the representation of collective artifacts.
– *Poor support for modeling dynamic features*: The dynamic manipulation of the process model in IMS LD is effected through "read" operations which return the state of process elements, and "write" operations which modify them (e.g., change-property-value, hide / show elements, and send notification). In [30] this requirement is posited as "more write operations need to be provided". To reformulate that statement, IMS LD has little support for effecting changes to the learning process model once that model participates in a "live" session. It is, as we shall see, one of the most prominent stumbling blocks when attempting to combine adaptivity in the specification under discussion.
– *Poor support for modeling complicated control flow*: A fourth major problem is how to model complex process structures. IMS LD provides the *play, act, rolepart*, and *activity-structure* elements to model structural relations at different levels. Primarily linear structured learning / teaching processes with concurrently executable activities can be modeled. However, as Caeiro et. al. [31] pointed out, while modeling network structures, the linear structure of a play with a series of *acts* introduced a great rigidity. Limited support to modeling non-linear structural relations among activities can be achieved through the use of custom conditions and notifications, likely at the expense of the comprehensibility and maintainability of the final result. From the perspective of adaptation, this is a major shortcoming, as it eliminates an entire category of potential adaptation determinants.

[2] Some representational facilities are available in IMSLD to support creation of groups (min-persons and max-persons) and although assignment of users to groups can be achieved, fully automatic on-the-fly creation of groups may require additional representational devices.

- *Poor support for modeling varied forms of social interaction*: As already mentioned, IMS LD uses a metaphor of a theatrical play to model learning / teaching processes. A play consists of a sequence of acts and within an act there is a set of role-parts. These role-parts can run together in parallel. Role-parts enable multiple learners, playing the same or different roles, to do the same thing or different things concurrently on the same act. If a group of people performs a synchronous activity, IMS LD enables them to use a conference service but provides no means at activity level to support collaboration. In collaborative learning processes, it is quite usual that people with the same or / and different roles perform a shared activity through direct or indirect interaction. While making the joint effort, people with different roles may have different rights to interact with other roles and the environment. In particular, it can not be clearly modeled by using IMS LD whether and how people collaborate, because people may work in a variety of social forms: Individually, in an informal group, in sub-groups, in a group as a whole, or in a community.
- *No exchange of information across Units of Learning*: This is a very limiting factor, as different UoLs can only "communicate" and interoperate through global properties, and employing entirely custom approaches.
- *Poor modeling of services and their characteristics*: The set of services that the specification includes by default is very limited and incorporate hardly any support for retrieving or settings advanced characteristics. In fact, the use of "tools" / "services" is seen as opaque from the perspective of the specification (i.e., the specification is not at all concerned about the inner workings of said tools and services, or how learners employ them). For instance, conference types are limited to "synchronous", "asynchronous" and "announcement"; this is clearly an immensely insufficient level of detail if adaptations are to be supported. Additional services can be "namespaced"[3] into the specification, but this is cumbersome technically and presumes that a de facto standard set of services and respective attributes emerges through bilateral agreements between specification adopters.
- *Acts within plays cannot be re-sequenced or structurally modified*: Although, formally speaking, this can be treated as a special case of the "poor support for modeling complicated control flow" mentioned earlier, it is detrimental enough for adaptivity that it is worth isolating and highlighting by itself. The IMS LD specification explicitly forbids changing the sequence of acts in a play, including doing so using conditions. It is also very difficult, if at all possible, to use a one-act-per-play approach, because in that scenario activities have to be independent of each other. Also, plays are handled differently per the specification, being made available concurrently (but without the possibility to express interconnections and interdependencies). Finally, it is also not possible to make activities available that are not already predefined in the act. Removing activities can be simulated by making them invisible though. This leads inevitably to having to exhaustively list all alternative configurations that can lead to the desired adaptive states through showing and hiding only, which quickly leads to an impractical (if at all possible) to manage combinatorial explosion of possible initial play configurations.

[3] Refers to the use of XML namespaces to create additions to an existing XML-based specification (like IMS LD).

– *Notifications cannot be added / removed (enabled / disabled) dynamically*: This simply means that the cause and effect relationships of learners' activities cannot be modified adaptively, based for example on the current collaboration context, or any other factors derived from the adaptation determinants. And this, again, is yet another lost opportunity to extend automated support to the learners.

It is important, at this point, to realize that all the above problems point directly to three characteristics of IMS LD: (a) absence of a run-time component of the specification (which describes what should happen and how at runtime, like the SCORM Runtime Environment specification [34]); (b) absence of an event model allowing for registration by event subscribers, and / or direct polling of individual event sources; (c) very limited support for modifying the collaboration process once a "run" of a UoL has started. These are arguably the main directions in which work needs to be directed in order to evolve the IMS LD specification in one that can fully support activity-oriented collaboration support in e-learning settings. To get a feeling about the effect that solving these issues will have, consider that solving items (a) and (b) above, would provide full bidirectional communication between the hosted learning resources and the runtime environment.

5 Conclusions

Based on the preceding section, we can now outline the extent to which IMS LD meets the adaptation requirements that were set forth in section 2 above. In short, the IMS LD specification is well thought out and allows for a considerable variety of dynamic modifications of learning activities (e.g., hiding and showing activities, environments, etc.) With some provisions, several of the basic mechanisms for adaptation at the hypertext level are possible. Nevertheless, the specification has important weaknesses which prevent its direct use in adaptive collaboration support scenaria at the moment. Some of these changes (e.g., the addition of a runtime environment specification) are major steps, and may require considerable effort and time to attain.

The results are summarized in the following table. It is hoped that the analysis elaborated upon herein, along with provided coarse outline of required research steps, will stimulate interesting discussions at the workshop, and spur additional work in this very new and very interesting field in the future.

Table 1. IMS LD vs. Requirements for adaptively supporting collaborative learning.

Id.	*Description*	*IMS LD support*	*Explanation*
A1	Maintain user- and learner-profile data	Yes	Through associated specifications that can be "collocated" – such as the IMS / LOM Metadata, and the IMS LIP specifications
A2	Maintain data about collaboration activity and activity of groups as collective entities	Minimal	Minimal information about tools and services employed by learners. No appropriate "companion" specifications. No explicit representation of groups.

Id.	*Description*	*IMS LD support*	*Explanation*
A3	Support algorithms / strategies that identify collaboration partners	Some	Conditions, expressions and notifications allow for a reasonably powerful embedded "if-then-else" style rules, with some limited event-based triggering as well. More involved algorithms have to be implemented externally and communicate their results through global variables
A4	Support alternative policies for / approaches to, group initiation	No	Even when using "roles" to simulate groups, users have to be assigned to roles at the beginning of a run, and assignment cannot be done automatically
B1	Support the modeling of groups (as entities)	Minimal	Groups cannot be directly modeled, except through the misappropriation of the "role" element
B2	Support the modeling of group activities	Some	Good support for learning activities of multiple individuals. But: no (real) groups – see above; no support for modeling artifacts; and, inflexible activity structures.
B3	Guide collaboration based on process specification	Some	No provisions for checking if or when expected activities take place (no runtime model, and no event model either).
B4	Maintain collaboration activity data	No	Not part of the specification at the moment. An amendment could specify how such data can be accessed, leaving it up to the runtime system how to collect and store it.
B5	Support the identification of group activity patterns in a semantically meaningful way	No	Not possible without group activity models and collaboration activity data.
B6	Support alternative algorithms / strategies to identify possible and appropriate interventions	Some	Same as A3, but here external algorithms (e.g., for pattern matching) more likely. Interventions can be identified, but most of them not really applied at the moment (other than simple hiding / showing, etc.)
B7	Enable adaptation algorithms to modify any aspect of the collaboration process	Minimal	The collaboration process specification can only be changed in very few ways once a run starts at the moment.

Acknowledgements. The work reported in this paper has been partially supported by the following projects: (a) Socrates Minerva "Adaptive Learning Spaces" (ALS) project (229714-CP-1-2006-1-NL-MPP) -- http://www.als-project.org; (b) "Adaptive Support for Collaborative E-Learning" (ASCOLLA) project, supported by the Austrian Science Fund (FWF; project number P20260-N15).

References

1. Pigliapoco, E., Bogliolo, A.: The effects of the Psychological Sense of Community in online and face-to-face academic courses. ICL2007 Conference, September 26 -28, Villach, Austria (2007).
2. Johnston, J., Killion, J., Oomen, J.: Student Satisfaction in the Virtual Classroom. The Internet Journal of Allied Health Sciences and Practice, 3(2) (2005).
3. Picciano, A.G.: Beyond student perceptions: issues of interaction, presence, and performance in an online course. Journal of Asynchronous Learning Networks (JALN), 6(1) (2002).
4. McCarthy, M.E., Pretty, G.M.H., Catano, V.: Psychological Sense of Community and Student Burnout. Journal of College Student Development, 31 (May), 211--216 (1990).
5. Frankola, K.: The E-learning taboo–high dropout rates: Best practices for increasing online course completion rates. Syllabus, June 2001, 14--16 (2001).
6. Kelsey, K. D., D'Souza, A.: Student motivation for learning at a distance: Does interaction matter? Online Journal of Distance Learning Administration, 7(2) (2004).
7. Haythornthwaite, C., Kazmer, M., Robins, J., and Shoemaker, S.: Making connections: Community among computer-supported distance learners. In: Proceedings of the Association for Library and Information Science Education 2000 Conference, San Antonio, TX.
8. IMS Global Learning Consortium: Learning Design Specification, http://www.imsglobal.org/learningdesign/
9. Paramythis, A., Loidl Reisinger, S.: Adaptive Learning Environments and eLearning Standards. Electronic Journal on e-Learning, 2(1), 181–194 (2004).
10. Wiley, D.: Learning Objects: Difficulties and Opportunities, http://wiley.ed.usu.edu/docs/lo_do.pdf (2003)
11. Strijbos, J.W., Kirschner P.A., Martens R.L., Dillenbourg, P. (Eds.): Computer-Supported Collaborative Learning; (Vol. 3). What we know about CSCL and implementing it in higher education. Norwell: Kluwer (2004).
12. Soller, A.: Adaptive collaboration support technology. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The adaptive web: Methods and strategies of web personalization. LNCS, vol. 4321, pp. 573—595. Springer, Heidelberg (2007).
13. Brooks, C., Hansen, C., Greer, J.: Social Awareness in the iHelp Courses Learning Content Management System. Proceedings of the Workshop on Social Navigation and Community-Based Adaptation Technologies, held in conjunction with the Adaptive Hypermedia and Adap-tive Web-Based Systems (AH'06) conference, June 20th, 2006, Dublin, Ireland (2006).
14. Vivacqua, A., Lieberman, H.: Agents to assist in finding help. In: Turner, T., Szwillus, G., Czerwinski, M., Patern`o, F. (eds.) Human factors in computing systems: CHI 2000 conference proceedings (pp. 65--72). New York: ACM (2000).
15. Terveen, L., McDonald, D.W.: Social matching: A framework and research agenda. ACM Transactions on Computer-Human Interaction, 12(3), 401--434 (2005).
16. Weber, G., Brusilovsky, P.: ELM-ART: An Adaptive Versatile System for Web-based Instruction. International Journal of Artificial Intelligence in Education, 12, 351--384 (2001).
17. Mitrovic, A., Suraweera, P., Martin, B., Weerasinghe, A.: Db-Suite: Experiences with three intelligent, web-based database tutors. Journal of Interactive Learning Research, 15(4), 409--432 (2004).
18. Brusilovsky, P., Peylo, C.: Adaptive and Intelligent Web-based Educational Systems. International Journal of Artificial Intelligence in Education, 13, 156–169 (2003)
19. Carro, R.M., Ortigosa, A., Schlichter, J.: Adaptive Collaborative Web-based Courses. In: Cueva, J.M., González, M., Joyanes, L., Labra, E., Paule, M.P. (eds.) Web Engineering. LNCS, vol. 2722, pp. 130--33. Springer, Heidelberg (2003)

20. Quignard, M., Baker, M.: Favouring modellable computer-mediated argumentative dialogue in collaborative problem-solving situations. In: Proceedings of the 9th International Conference on AI in Education, pp. 129– 136. IOS Press, Amsterdam (1999)
21. IMS Global Learning Consortium: Learner Information Package Specification, http://www.imsglobal.org/profiles/
22. Vizcaino, A., Contreras, J., Favela, J., Prieto, M.: An adaptive, collaborative environment to develop good habits in programming. In: Proceedings of the 5th International Conference on Intelligent Tutoring Systems, pp. 262--271. Montreal, Canada (2000)
23. Masthoff, J.: Group modeling: Selecting a sequence of television items to suit a group of viewers. User Modeling and User Adapted Interaction, 14, pp37--85. Springer-Verlag (2004).
24. Koper, E.J.R.:Modeling Units of Study from a Padagogical Perspective: the Pedagogical Meta-model behind EML. Educational Technology Expertise Centre Open University of the Netherlands (2001)
25. Koper, E.J.R., Olivier. B.: Representing the Learning Design of Units of Learning. Educational Technology & Society. 7(3), p.97-111 (2004).
26. Specht, M., Burgos, D.: Implementing Adaptive Educational Methods with IMS Learning Design. In: ADALE Workshop Proceedings of AH2006 Conference (2006). Retrieved May 2nd, 2008 from http://hdl.handle.net/1820/718
27. Burgos, D., Tattersall, T., Koper, R.: Representing adaptive eLearning strategies in IMS Learning Design. In: Proceedings of the 2006 TENCompetence Conference, Sofia, Bulgaria (2006).
28. IMS Global Learning Consortium: Question & Test Interoperability Specification, http://www.imsglobal.org/question/
29. Hauger, D., Köck, M.: State of the Art of Adaptivity in E-Learning Platforms. In: Proceedings of the 15th Workshop on Adaptivity and User Modeling in Interactive Systems, 24.-26.9.2007, Halle/Saale, Germany, pp. 355-360 (2007).
30. Miao, Y., Hoeksema, K., Hoppe, H. U., and Harrer, A.: CSCL scripts: modelling features and potential use. In: Proceedings of the 2005 Conference on Computer Support For Collaborative Learning: Learning 2005: the Next 10 Years! (Taipei, Taiwan, May 30 - June 04, 2005). Computer Support for Collaborative Learning. International Society of the Learning Sciences, 423—432 (2005).
31. Caeiro, M., Anido, L., Llamas, M.: A Critical Analysis of IMS Learning Design. In: Proceedings of CSCL 2003, p.363—367 (2003).
32. Hernandez, D., Asensio, J.I., Dimitriadis, Y.: IMS Learning Design Support for the Formalisation of Collaborative Learning Flow Patterns. In: Proceedings of the 4th International Conference on Advanced Learning Technologies (Aug.30 - Sep. 1, 2004, Joensuu, Finland), pp.350-354. IEEE Press (2004)
33. O'Donnell, A. M., Dansereau, D. F.: Scripted Cooperation in Student Dyada: A Method for Analyzing and Enhancing Academic Learning and Performance. In: Hertz-Lazarowitz, R., Miller, N. (eds.), Interaction in Cooperative Groups: The theoretical Anatomy of Group Learning (pp. 120--141). London: Cambridge University Press (1992).
34. Advanced Distributed Learning Initiative: SCORM 2004 3rd Edition Run-Time Environment, Version 1.0, http://www.adlnet.gov/scorm/20043ED/Documentation.aspx